MODELAGEM DE BANCO DE DADOS RELACIONAL

Uma abordagem prática e objetiva

Tássio Gonçalves

@tassiogoncalvesg

QUEM SOU EU?

TÁSSIO GONÇALVES

- Mestrando em Informática pela UFAL
- Esp. em Informática e Comunicação na Educação
- Bacharel em Sistemas de Informação pela UFAL
- Professor Tutor da UFAL
- Técnico em Laboratório de Informática do IFBA - Campus de Paulo Afonso.
- Professor no CETEP-Itaparica

- Saiba mais: https://tassiogoncalves.com.br/sobre

CONTEÚDO

PARTE I | INTRODUÇÃO E CONCEITOS

ROTEIRO

Banco de Dados
- Aplicações
- Definições

Sistema de Gerenciamento de Banco de Dados
- Definições
- Exemplos
- Vantagens
- Facilidades

Considerações Finais

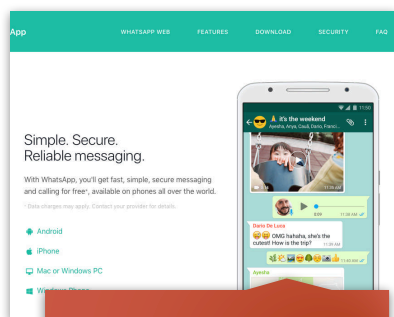
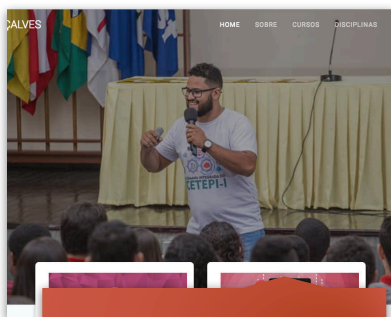
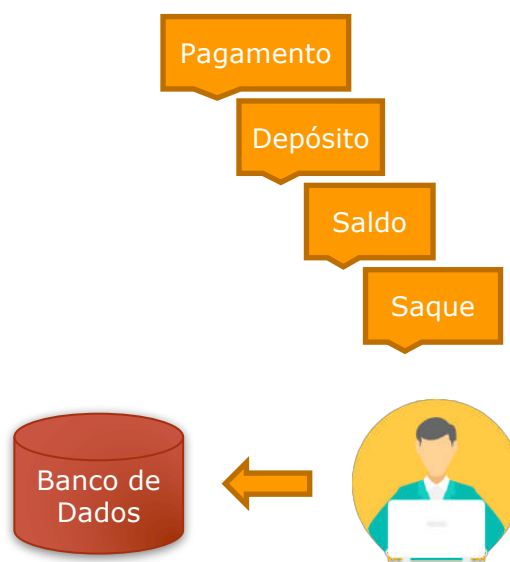
Referências

O QUE É UM BANCO DE DADOS?

Banco de Dados

- É uma coleção de dados inter-relacionados, representando informações sobre um domínio específico. (Navathe)
- É um conjunto de dados estruturados que são confiáveis, coerentes e compartilhados por usuários que têm necessidade de diferentes informações. (Fonseca)

- Podemos exemplificar situações clássicas como uma lista telefônica, um catálogo de livros ou um sistema de controle de RH de uma empresa.

EXEMPLOS DE APLICAÇÕES QUE FAZEM USO DE BANCO DE DADOS - (BD)

7

MAIS EXEMPLOS...

BANCO DE DADOS DE UMA AGÊNCIA

Coleção de dados

Clientes, ruas, cidades, datas de nascimento, contas, saldos.

Informações

1 – Neymar e Cristiano possuem um saldo superior a R$: 3mi;
2 – Abel e Gabigol moram na mesma cidade;
3 – Abel faz aniversário no dia 01 de setembro.

Domínio específico

Clientes de uma Agência Bancária

nome	cidade	nascimento	conta	saldo
Gabigol	Rio de Janeiro	1996-08-30	901	2580004.00
Neymar Jr	Paris	1992-02-05	559	509088070.00
Cristiano Ronaldo	Lisboa	1985-02-05	308	49508600.00
Abel Braga	Rio de Janeiro	1952-09-01	180	1760890.00

BANCO DE DADOS DE UMA FACULDADE

Coleção de dados

Alunos, números, classes, curso.

Informações

1 – Aristóteles e Platão são colegas de classe;
2 – João, Aristóteles e Platão fazem o mesmo curso;

Domínio específico

Alunos de uma Faculdade

cod	nome	numero	classe	curso
1	João	26	1	info
2	Aristóteles	5	2	info
3	Rafael	15	1	eng_ele
4	Platão	8	2	info

ESTRUTURA DE UM BANCO DE DADOS

Esquema do banco de dados **Projeto Geral do banco de dados**

Instâncias do banco de dados **Conjunto de informações contidas em determinado banco de dados, em um dado momento**

	nome	cidade	telefone	conta	saldo
Instância 1	Tássio	Paulo Afonso	75988880909	90	1200.00
Instância 2	Mércia	Glória	75988889999	30	4300.00
Instância 3	Maria	Petrolândia	87999989977	201	9800.00
Instância 4	Chicão	Paulo Afonso	75988009090	75	5800.00
Instância 5	Fátima	Delmiro Golveia	82981880754	300	10900.00

Esquema

USUÁRIOS TEM NECESSIDADES DIFERENTES

Cliente

- Qual o saldo da minha conta?

Gerente do Banco

- Quais clientes estão com saldo abaixo do limite?
- Onde moram os clientes com saldo superior a R$: 8.000,00?

nome	cidade	telefone	conta	saldo
Tássio	Paulo Afonso	75988880909	90	1200.00
Mércia	Glória	75988889999	30	4300.00
Maria	Petrolândia	87999989977	201	9800.00
Chicão	Paulo Afonso	75988009090	75	5800.00
Fátima	Delmiro Golveia	82981880754	300	10900.00

CONSULTAS E ATUALIZAÇÕES

Consultas

- Qual o time que mais vezes ganhou o Campeonato Carioca?
- Quantos times já conquistaram o título do Campeonato Carioca ao menos uma vez?
- Quantos títulos o América conquistou?

time	titulos
Fluminense	31
Botafogo	21
Flamengo	34
Vasco	24
Bangu	2
América	7
São Cristóvão	1

CONSULTAS E ATUALIZAÇÕES

Consultas

- Quantos alunos estão matriculados na disciplina Banco de Dados do CETEPI no semestre 2018.1?
- Quantos alunos se matricularam na disciplina Banco de Dados do CETEPI no ano de 2017?

disciplina	ano	semestre	alunos
APS	2019	1	16
BD	2019	1	128
VISUAL	2019	1	16
BD	2018	1	33
APS	2018	1	25

CONSULTAS E ATUALIZAÇÕES

Atualizações

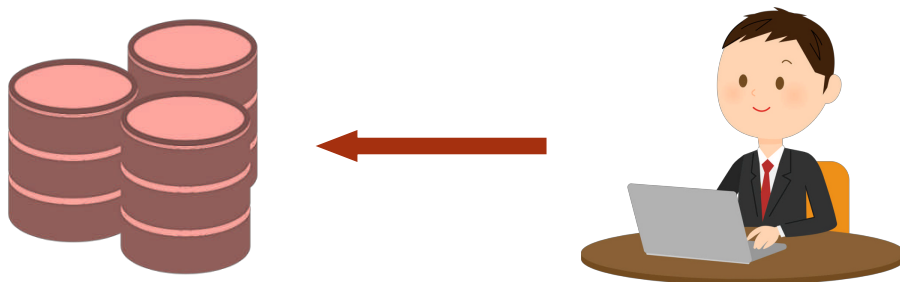
- Reajuste o salário dos programadores em 50%.
- Altere o nome de Juca para Juca da Silva.

nome	codigo	salario	cargo
José	2	2500.00	Engenheiro
Juca	18	1300	Programador
Sócrates	15	2800	Analista
Platão	79	1200	Professor
Aristócrates	46	1300	Programador

BANCO DE DADOS

- Pode ser gerenciado por um Sistema de Gerenciamento de Banco de Dados (SGBD)
  - Armazenamento, acesso e atualização.

SISTEMA DE GERENCIAMENTO DE BANCOS DE DADOS (SGBD)

O SGBD é um software que possui recursos capazes de manipular as informações do banco de dados e interagir com o usuário.

- É uma coleção de programas que permite aos usuários criar e manter um Banco de Dados. (Navathe)
- É constituído por um conjunto de dados (BD) associados a um conjunto de programas para acesso a esses dados. (Silberschatz)

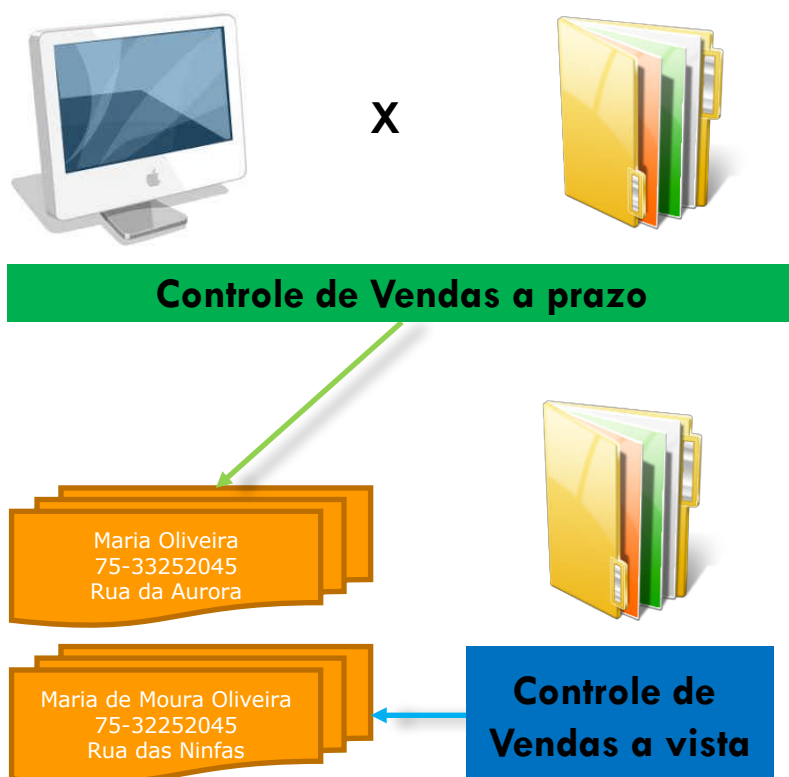
SGBD

Exemplos:

SGBD X SISTEMA DE ARQUIVOS

- Velocidade x Demora
- Facilidade x Dificuldade de acesso

Exemplo: atualizar os dados de José

SGBD X SISTEMA DE ARQUIVOS

O Sistema de arquivos terá consequentemente:

- Inconsistência
- Duplicação de esforços
- Gastos com armazenamento

Além de Redundância:

- Mesma informação pode ser repetida em mais de um arquivo.

Dificuldades como por exemplo: promoção "quem compra mais no mês"

VANTAGENS DOS SGBD

Controle de Vendas a prazo

- Velocidade
- Facilidade de acesso
- Redução da redundância
- Evita-se inconsistência

FACILIDADES DOS SGBDS

- Segurança
- Recuperação após falhas
- Restrições de Integridade
- Controle de Concorrência

FACILIDADES DOS SGBDS

nome	cidade	conta	saldo
Tássio	Paulo Afonso	90	1200.00
Mércia	Glória	30	4300.00
Maria	Petrolândia	201	9800.00
Chicão	Paulo Afonso	75	5800.00
Fátima	Delmiro Golveia	300	10900.00

Segurança:

- Permite a definição de diferentes níveis de permissão

- Exemplo bancário: tabelas de clientes e contas
  - **Caixa:**
    - pode consultar e depositar em todas as contas.
  - **Cliente:**
    - pode apenas consultar o saldo e sacar de sua conta (se houver saldo suficiente).

FACILIDADES DOS SGBDS

Segurança:

- Provê backup
  - Evitando perdas acidentais

    - Exemplo: Maria foi deletada!

nome	cidade	conta	saldo
Tássio	Paulo Afonso	90	1200.00
Mércia	Glória	30	4300.00
Chicão	Paulo Afonso	75	5800.00
Fátima	Delmiro Golveia	300	10900.00

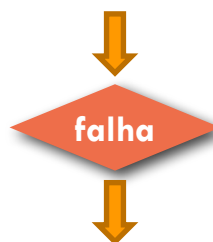
    - Exemplo: Maria foi recuperada!

nome	cidade	conta	saldo
Tássio	Paulo Afonso	90	1200.00
Mércia	Glória	30	4300.00
Maria	Petrolândia	201	9800.00
Chicão	Paulo Afonso	75	5800.00
Fátima	Delmiro Golveia	300	10900.00

FACILIDADES DOS SGBDS

Recuperação após falhas:

- O BD deve permanecer em um estado consistente após uma queda de energia ou queda do sistema

- Exemplo: **Tentativa de efetuar uma recarga de R$: 12,00**

FACILIDADES DOS SGBDS

Restrições de integridade:

- Faz a validação dos dados do BD.
- **Exemplo:** nota >=0 e <=10

matricula	nome	cod_disciplina	cod_curso	nota
5	José	112	info	7.0
8	João	119	seg	6.5
19	Platão	95	info	5.0
15	Sócrates	102	info	5.0
2	Aristóteles	143	info	8.0

- **Exemplo:** salário >= 980

nome	codigo	salario	cargo
José	2	2500.00	Engenheiro
Juca	18	1300	Programador
Sócrates	15	2800	Analista
Platão	79	1200	Professor
Aristócrates	46	1300	Programador

FACILIDADES DOS SGBDS

Restrições de integridade:

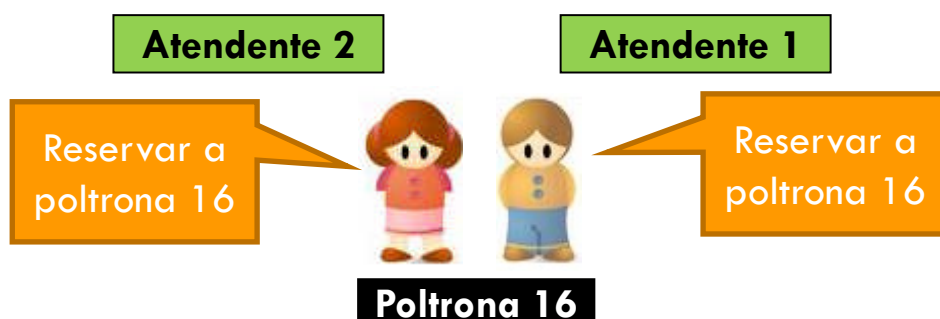
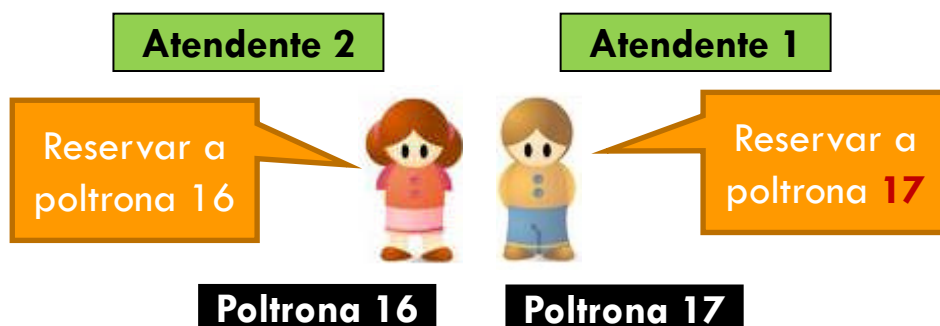
- Mais exemplos: código do funcionário deve ser único e não nulo.

nome	codigo	salario	cargo
José	2	2500.00	Engenheiro
Juca	18	1300	Programador
Sócrates	15	2800	Analista
Platão	79	1200	Professor
Aristócrates	46	1300	Programador

FACILIDADES DOS SGBDS

Controle de concorrência: permite que vários usuários acessem o banco de dados simultaneamente de **forma coerente**.

Exemplo:

- Quando muitos atendentes tentam reservar o mesmo lugar em um determinado voo.

- O que deve acontecer?
  - Nesse caso o atendente que finalizar o atendimento primeiro o cliente ficará com a poltrona 16 e o próximo poderá escolher outra, por exemplo a 17.

CONSIDERAÇÕES FINAIS

- SGBDs apoiam o gerenciamento de grandes volumes de dados.
- Os benefícios incluem a recuperação de falhas, acesso concorrente, desenvolvimento simplificado e ágil, integridade dos dados e segurança.
- Níveis de abstração proporcionam independência dos dados.
- Administradores de banco de dados (DBAs) possuem cargos importantes, mas são bem pagos!

QUANDO NÃO USAR SGBDS Relacionais:

- Se há requisitos críticos de tempo real que nenhum SGBD possa atender, nesse caso deve-se procurar uma solução própria ou outras arquiteturas.
- Se os SGBDs relacionais existentes não são capazes de lidar com a natureza dos dados da aplicação
- Se os usuários necessitam de operações específicas que não são oferecidas pelos SGBDs disponíveis.

REFERÊNCIAS

- Exemplos: Notas de Aula de Introdução a Banco de Dados, Fonseca, F. Centro de Informática – UFPE.

- Sistemas de Banco de Dados, Elmasri, Ranmez e Navathe, Shamkant B., Pearson, 4ª edição.

- Sistema de Banco de Dados, Silberschatz, A., Makron Books, 3ª edição.

EXERCÍCIO

1. Defina:
   a. Dados;
   b. Banco de Dados;
   c. Informação;
   d. SGBD;
2. O que são instancias em Banco de Dados?
3. Cite 4 Exemplos de SGBD.
4. O que é esquema em BD?
5. Exemplifique quais coleções de dados podemos ter para o domínio CETEP, dê exemplos de informações que podemos obter dessas coleções.
6. Quais são as facilidades em utilizar um SGBD?
7. Explique Controle de Concorrência.

PARTE II

MODELAGEM DE DADOS

ROTEIRO

Modelagem de Dados

Modelo Entidade-Relacionamento

- Componentes
- Tipos de atributos
- Atributo em um relacionamento
- Cardinalidade
- Auto relacionamento
- Relacionamento Ternário
- Herança
- Entidade Fraca

Considerações Finais

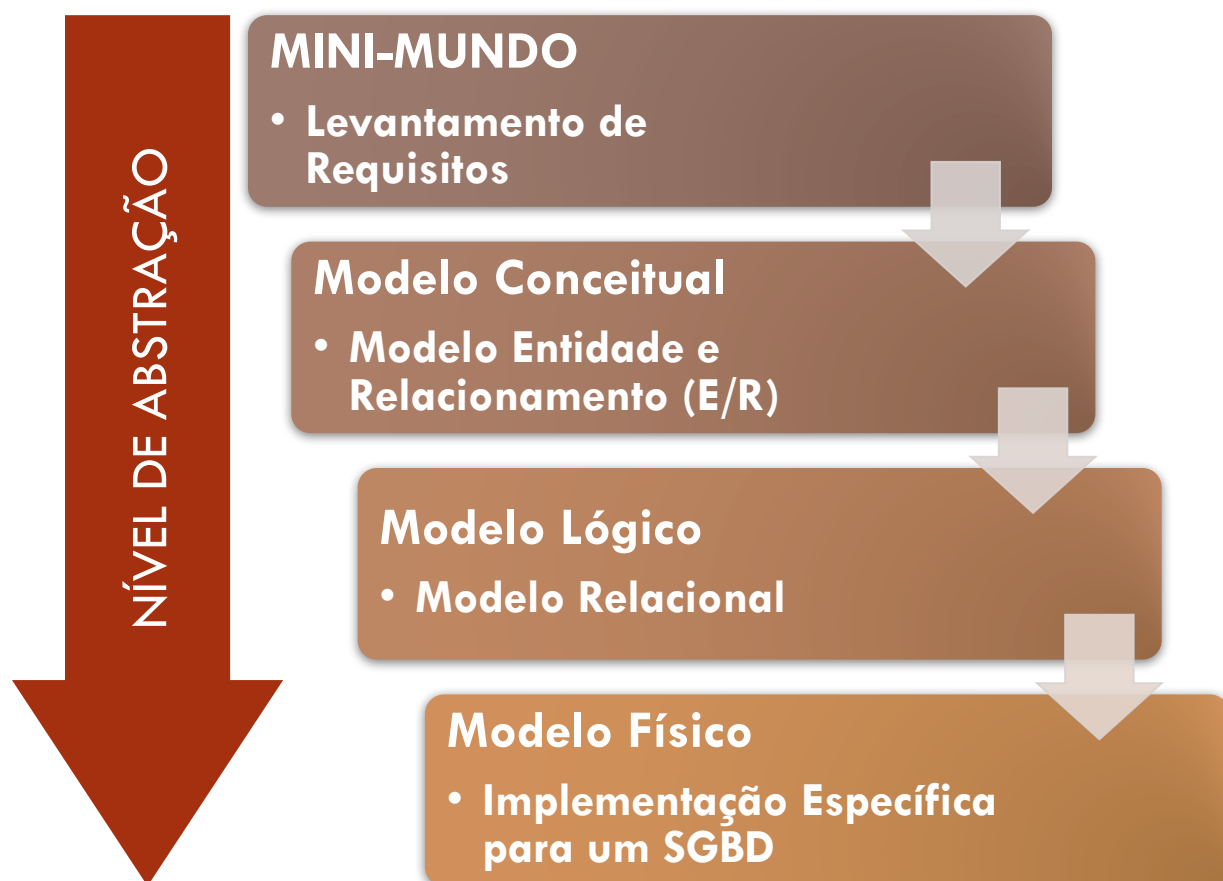
MODELAGEM DE DADOS

NÍVEL DE ABSTRAÇÃO

MINI-MUNDO
- Levantamento de Requisitos

Modelo Conceitual
- Modelo Entidade e Relacionamento (E/R)

Modelo Lógico
- Modelo Relacional

Modelo Físico
- Implementação Específica para um SGBD

MODELOS CONCEITUAIS, LÓGICOS E DE DADOS FÍSICOS

Modelos ER e modelos de dados são normalmente desenhadas em até três níveis de detalhe:

Modelo de dados conceitual: a visão de mais alto nível que contém o mínimo de detalhe. Seu valor é de mostrar um âmbito geral do modelo e retratar a arquitetura do sistema. Para um sistema de alcance menor, pode não ser necessário desenhar. Em vez disso, comece com um modelo lógico.

Modelo de dados lógico: contém mais detalhes que um modelo conceitual e entidades operacionais e transacionais mais detalhadas agora são definidas. O modelo lógico é independente da tecnologia na qual ele será implementado.

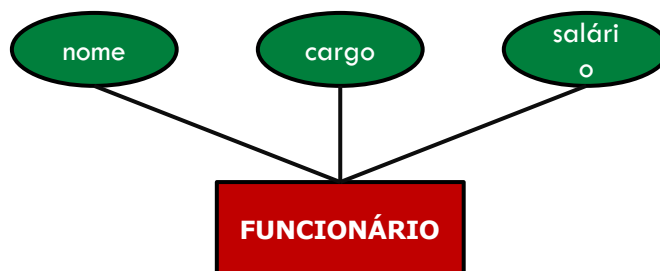
Modelo de dados físico: um ou mais modelos físicos podem ser desenvolvidos a partir de cada modelo lógico. Modelos físicos devem mostrar detalhes de tecnologia suficientes para produzir e implementar o banco de dados.

MODELO ENTIDADE-RELACIONAMENTO

- Em engenharia de software, um modelo entidade relacionamento (modelo ER) é um modelo de dados para descrever os dados ou aspectos de informação de um domínio de negócio ou seus requisitos de processo, de uma maneira abstrata que em última análise se presta a ser implementada em um banco de dados, como um banco de dados relacional.
- O "MER" foi desenvolvido por Peter Chen e publicado em um artigo de 1976, é um modelo baseado na percepção do mundo real, que consiste em um conjunto de objetos básicos chamados entidades e nos relacionamentos entre esses objetos.
- Facilita o projeto de banco de dados, possibilitando a especificação da estrutura lógica geral do BD.

COMPONENTES DO DIAGRAMA E-R (PETER CHEN)

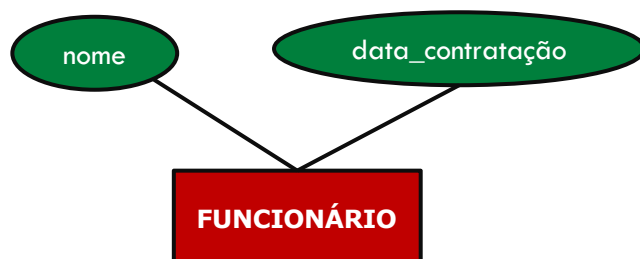
- Retângulos: representam as entidades

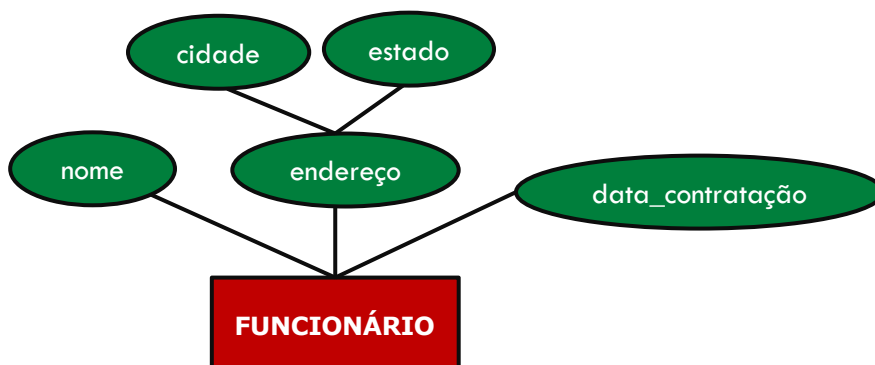
- Elipses: representam atributos

- Losangos: representam os relacionamentos

- Linhas: ligam atributos entidades e entidade a relacionamentos

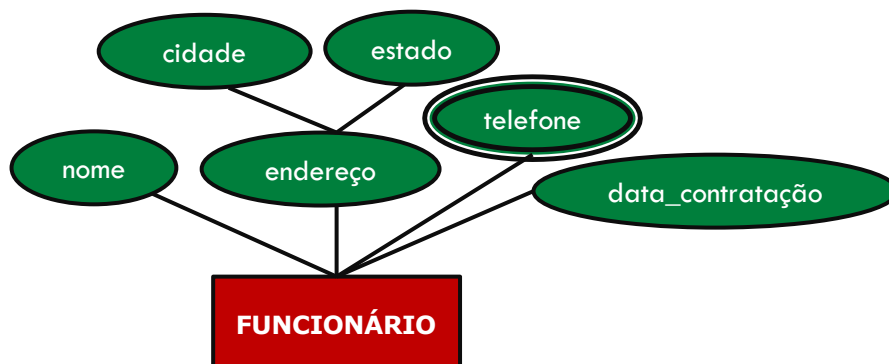
TIPOS DE ATRIBUTOS

Atributo Monovalorado: assume um único valor para cada elemento da entidade

- Ex: Nome e Data da Contratação

Atributo Composto: formado por um ou mais sub atributos

- Ex: Endereço

TIPOS DE ATRIBUTOS

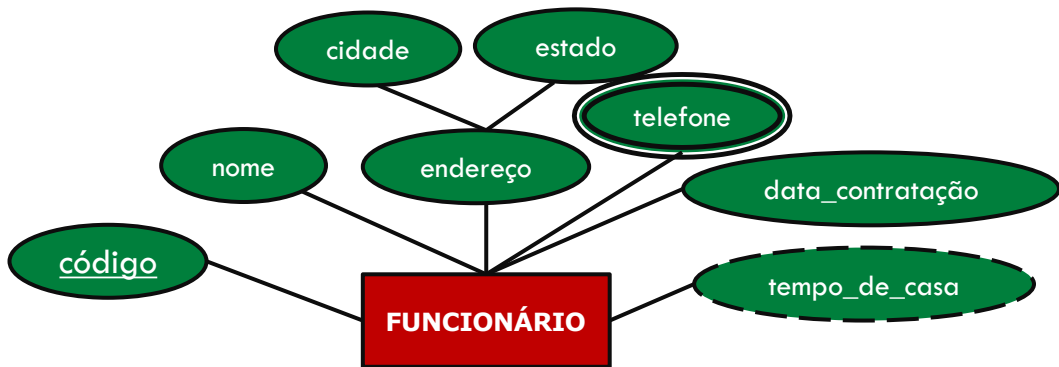
Atributo Multivalorado: uma única entidade tem diversos valores para este atributo

- Ex: Telefone

Atributo Derivado: o valor deste tipo de atributo pode ser derivado de outros atributos a ele relacionados.

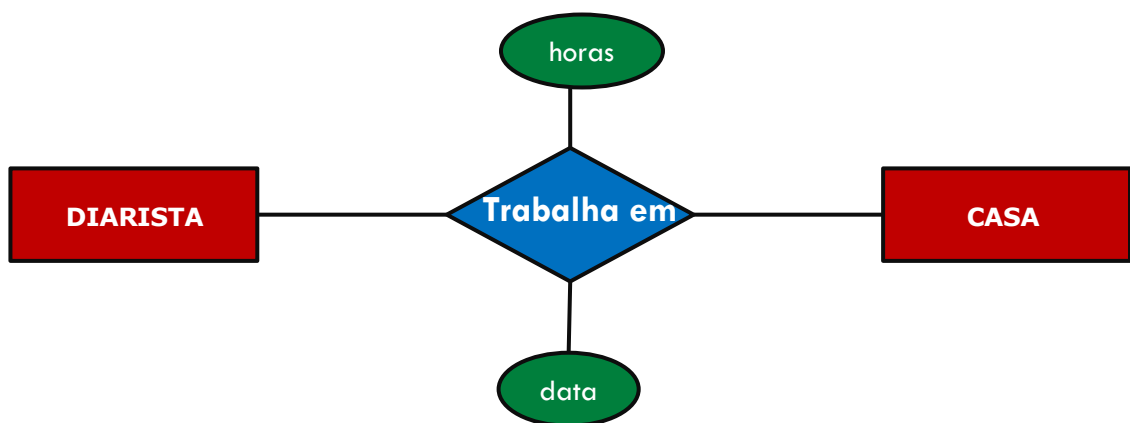
- Ex: Tempo de casa

TIPOS DE ATRIBUTOS

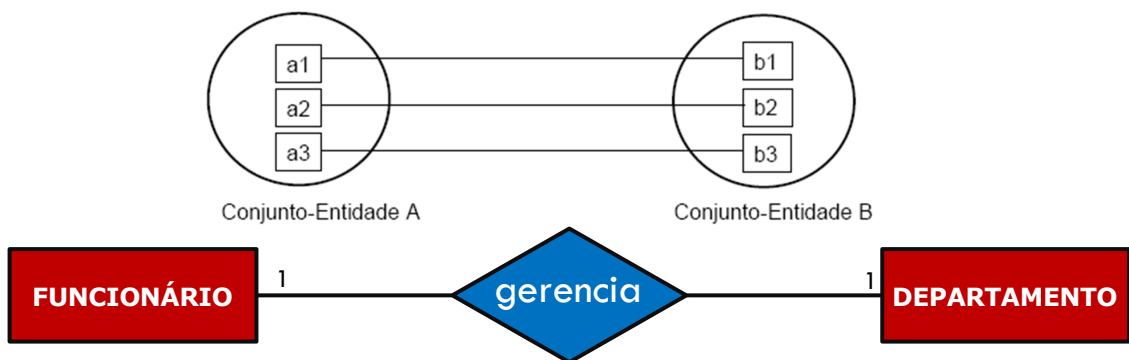
Atributo Determinante: identificador de uma entidade (também conhecido como atributo chave)

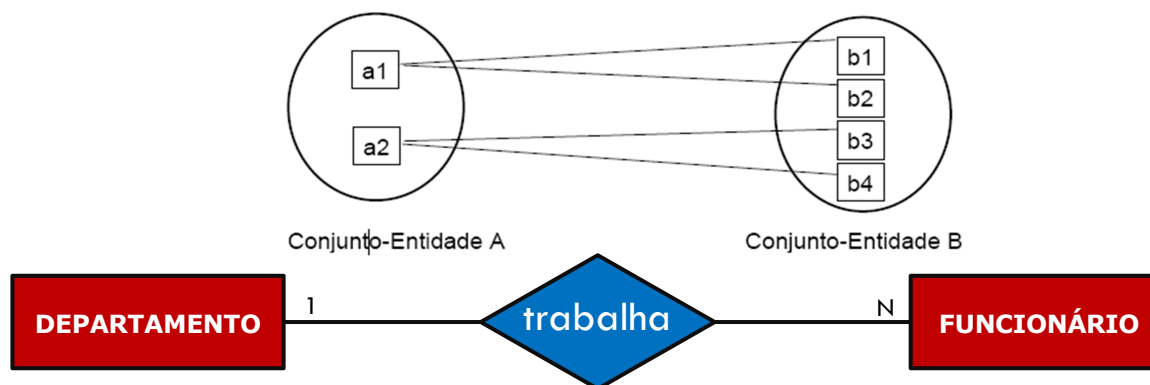
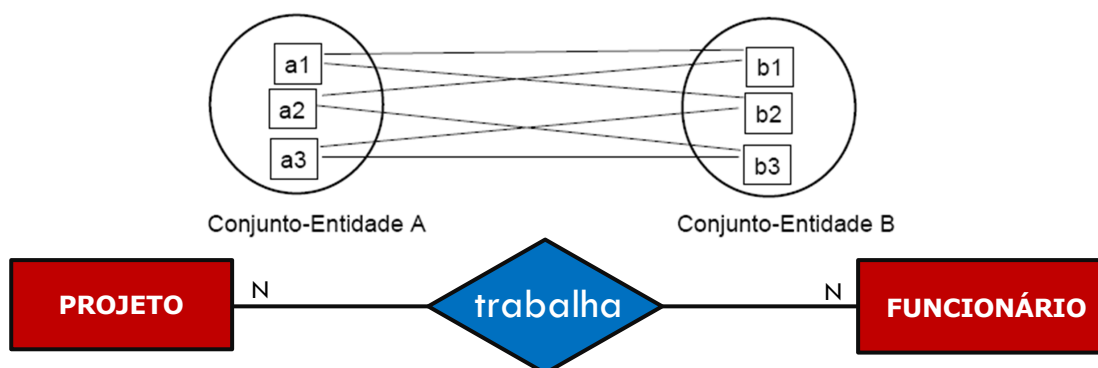
- Ex: Código

ATRIBUTOS

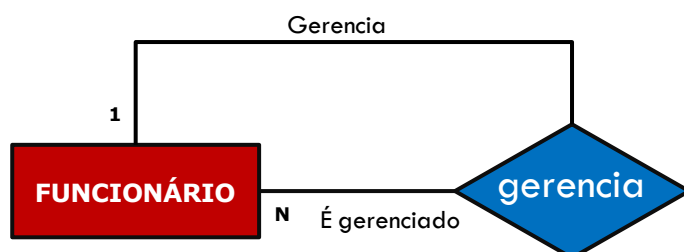
Atributos em um relacionamento: algumas vezes podemos encontrar atributos em determinados relacionamentos

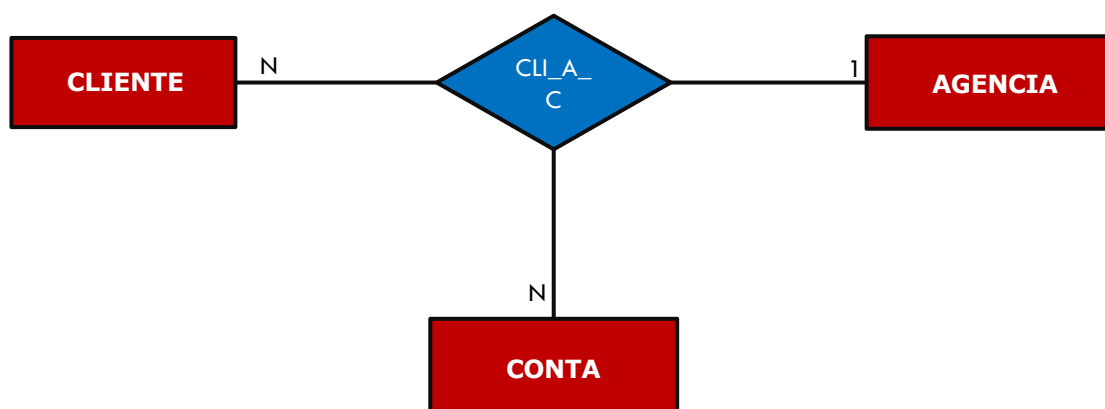
- A central de diaristas mantém o cadastro de suas diaristas por matrícula, nome, endereço, telefone e data de nascimento.
- As diaristas trabalham em várias casas.
- Tais casas são caracterizadas pelo código e endereço.
- Deseja-se guardar a data do serviço e o número de horas que uma diarista trabalha em uma casa.

CARDINALIDADE

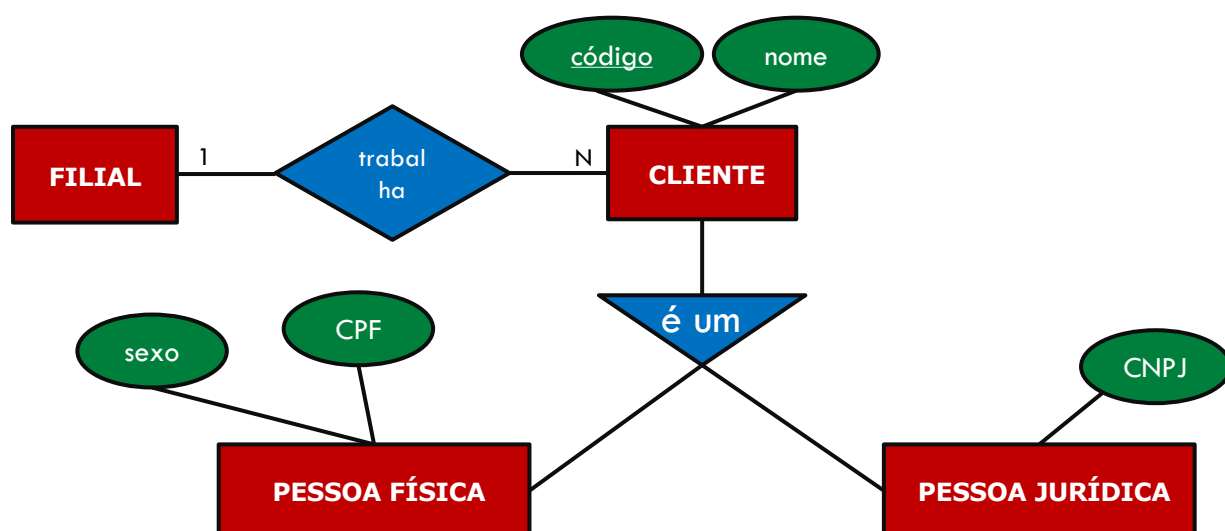
Em modelagem de dados a cardinalidade é um dos princípios fundamentais sobre relacionamento de um banco de dados relacional. Nela são definidos o graus de relação entre duas entidades ou tabelas.[1]
No modelo relacional, podemos ter os seguintes níveis de relacionamento: 1:N, N:N, 1:1.

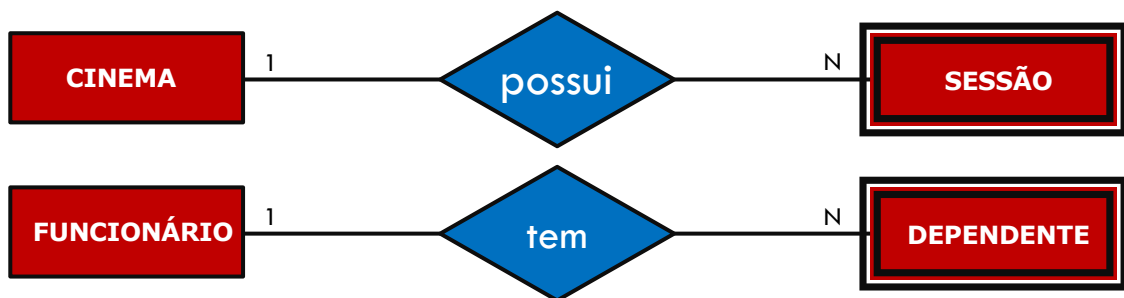
Um-para-um (1:1): Uma entidade em A está associada no máximo a uma entidade em B e uma entidade em B está associada no máximo a uma entidade em A

CARDINALIDADE

Um-para-muitos (1:N): Uma entidade A está associada a qualquer número de entidades em B, enquanto uma entidade em B está associada no máximo a uma entidade em A

Muitos-para-muitos (N:N): Uma entidade em A está associada a qualquer número de entidades em B, e uma entidade em B está associada a qualquer número de entidades em A.

AUTO RELACIONAMENTO

Relacionamento que ocorre de uma entidade com ela mesma.

RELACIONAMENTO TERNÁRIO

Relacionamento que ocorre entre três entidades, pode existir também com mais de três entidades, é conhecido como de Relacionamento N-ário (Para cada relacionamento n-ário [ternário, quaternário, etc.], onde n > 2).

HERANÇA

- Podemos Generalizar / Especializar uma Entidade dentro de um modelo E-R
- Podemos dividir em categorias cada entidade

ENTIDADE FRACA

- Ocorre quando a Entidade não possui sequer identidade própria, sendo sua identificação composta pela chave proveniente da entidade dona concatenada.

CONSIDERAÇÕES FINAIS

Revisando o Modelo E-R

- Principais componentes
- Tipos de atributos
- Conceitos chave
  - Atributo em um relacionamento
  - Cardinalidade
  - Auto relacionamento
  - Relacionamento Ternário
  - Herança
  - Entidade Fraca

EXERCÍCIOS

1 - Faça a Modelagem E-R da seguinte situação:

- A universidade mantém o cadastro de seus alunos por matricula, nome, endereço, fone e data de nascimento.
- Os alunos cursam as disciplinas que são oferecidas na universidade.
- Tais disciplinas são caracterizadas pelo código, título, descrição e carga horária.
- Deseja-se guardar a nota obtida e o ano em que um aluno cursa uma disciplina.

EXERCÍCIOS

2 - Faça a Modelagem E-R da seguinte situação:

CASO - FLORICULTURA

Uma floricultura deseja informatizar suas operações. Inicialmente, deseja manter um cadastro de todos os seus clientes, mantendo informações como: RG, nome, telefone e endereço. Deseja também manter um cadastro contendo informações sobre os produtos que vende, tais como: nome do produto, tipo (flor, vaso, planta,...), preço e quantidade em estoque. Quando um cliente faz uma compra, a mesma é armazenada, mantendo informação sobre o cliente que fez a compra, a data da compra, o valor total e os produtos comprados.

EXERCÍCIOS

3 - Faça a Modelagem E-R da seguinte situação:

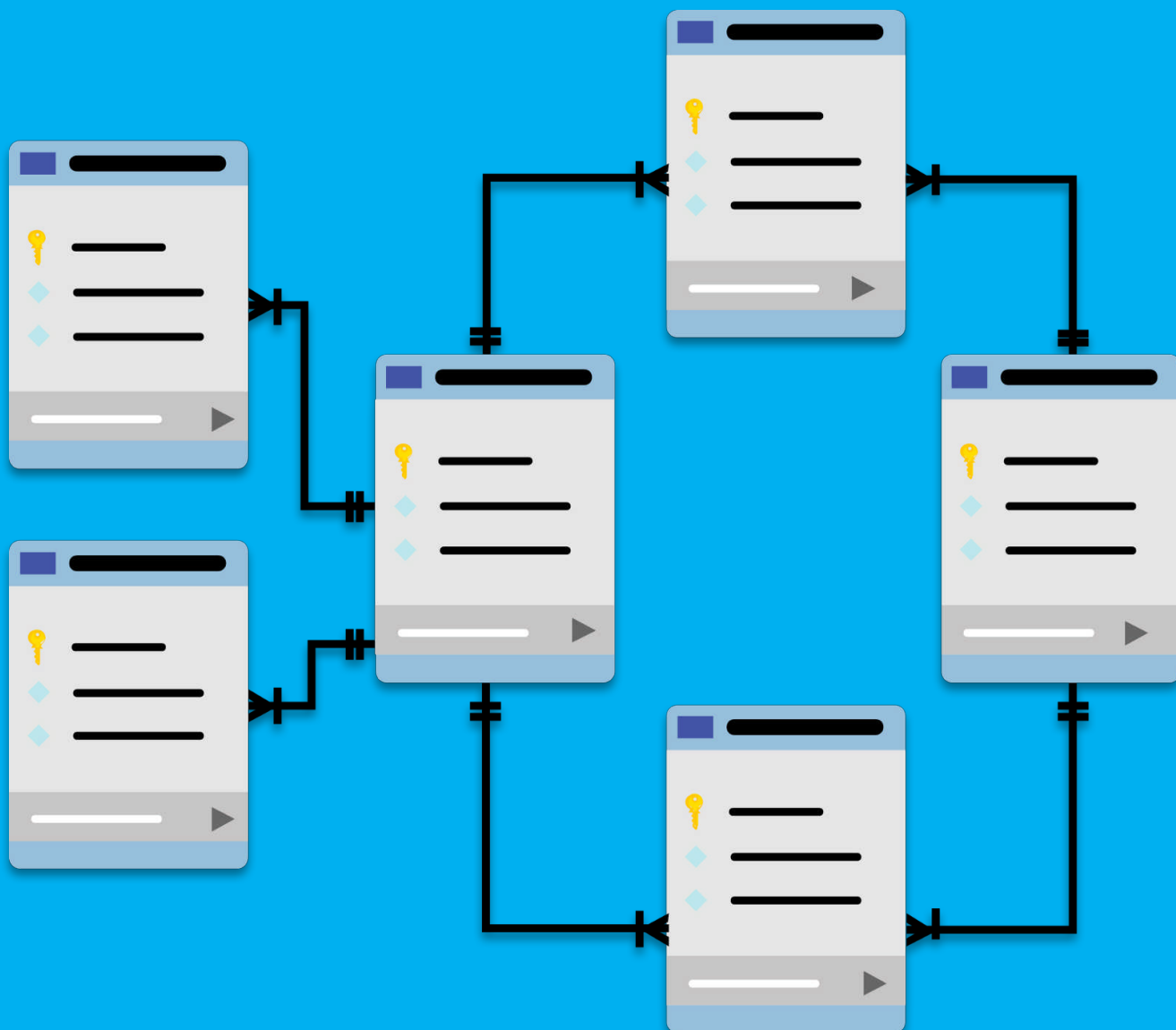
CASO – BERÇÁRIO

Um berçário deseja informatizar suas operações. Quando um bebê nasce, algumas informações são armazenadas sobre ele, tais como: nome, data do nascimento, peso do nascimento, altura, a mãe deste bebê e o médico que fez seu parto. Para as mães, o berçário também deseja manter um controle, guardando informações como: nome, endereço, telefone e data de nascimento. Para os médicos, é importante saber: CRM, nome, telefone celular e especialidade.

EXERCÍCIOS

4 - Faça a Modelagem E-R da seguinte situação:

CASO – ESCOLA

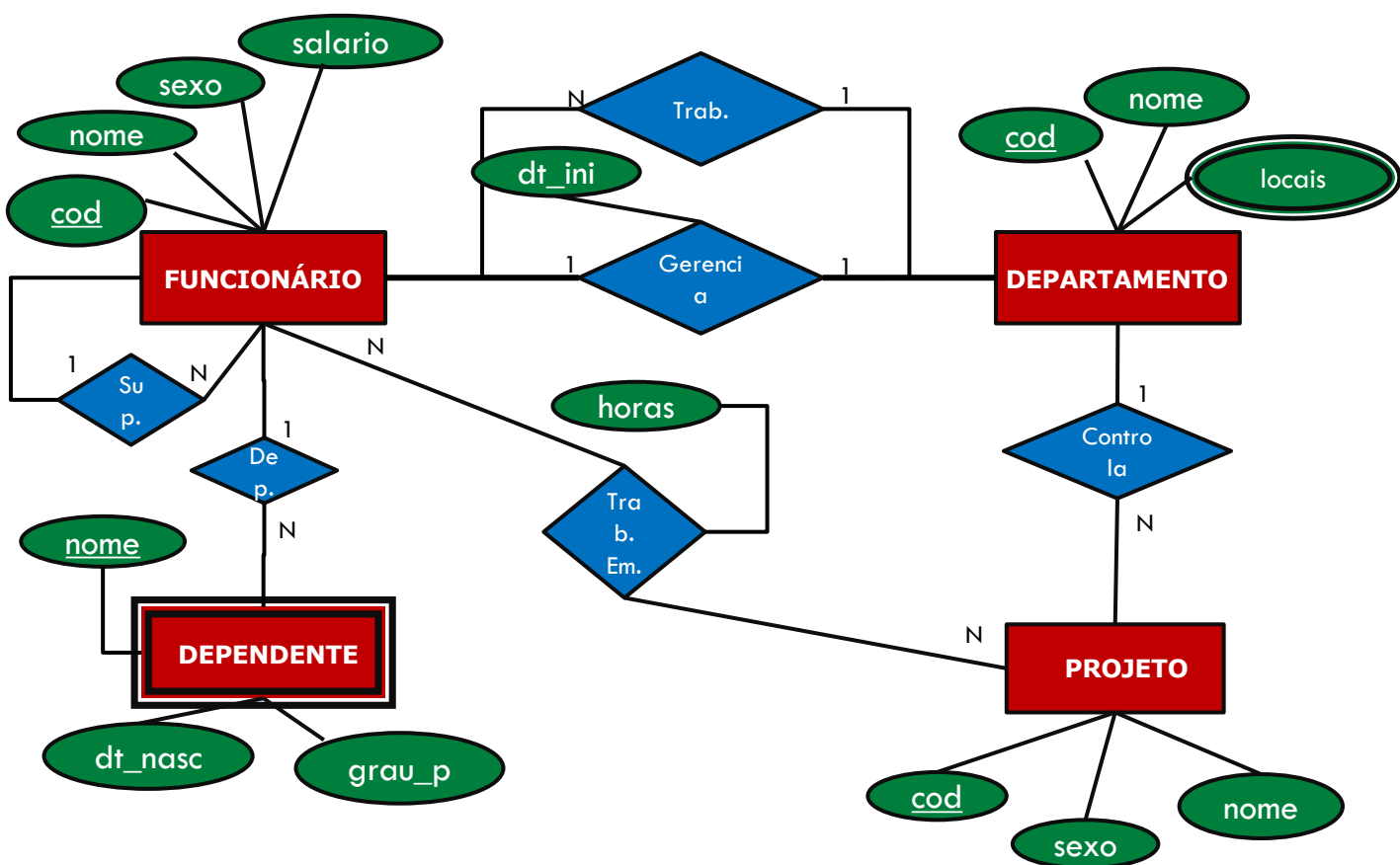
Uma Escola tem várias turmas. Uma turma tem vários professores, sendo que um professor pode ministrar aulas em mais de uma turma. Uma turma tem sempre aulas na mesma sala, mas uma sala pode estar associada a várias turmas (com turnos diferentes).

PARTE III
ESQUEMA
RELACIONAL

ROTEIRO

- **ER / Relacional**
- **Esquema Relacional**

ER / RELACIONAL

Um esquema relacional pode ser facilmente derivado de um esquema conceitual desenvolvido usando o modelo E-R

Seja o exemplo de modelagem E-R a seguir

ER / RELACIONAL

O modelo físico ou modelo relacional é o modelo utilizado para mapear um esquema de modelo ER para um conjunto de tabelas inter-relacionadas (esquema de banco de dados relacional). Veremos aqui um passo a passo para fazer corretamente este mapeamento.

Passo 1

Para cada **Entidade Regular E** no esquema ER, criar uma relação R que inclui todos os atributos de R.

FUNCIONÁRIO	cod	nome	sexo	salario

ER / RELACIONAL

Passo 2

Para cada **Entidade Fraca W** no esquema E-R que tenha como entidade proprietária E,

- criar uma relação R e incluir todos os atributos de W como atributos de R,
- incluir o(s) atributo(s) chave primária da relação proprietária E
- A chave primária de R é a combinação da chave primária de E e da chave parcial de W

DEPENDENTE	cod_FK	nome	dt_nasc	grau_p

ER / RELACIONAL

Passo 3

Para cada **Relacionamento R de 1:1** no esquema ER,

- identificar as relações das entidades participando do relacionamento R,
- escolher uma das relações e incluir como **chave estrangeira**, a chave primária da outra relação,
- incluir todos os atributos do relacionamento na relação escolhida

DEPARTAMENTO	cod	nome	cod_gerente_FK	dt_ini

ER / RELACIONAL

Passo 4

Para cada **Relacionamento R de 1:N** no esquema ER,

- identificar a relação S que representa a entidade do lado N,
- incluir como chave estrangeira a chave primária da relação que representa a entidade do lado 1,
- incluir os atributos do relacionamento em S.

FUNCIONÁRIO	cod	nome	sexo	salario	cod_dep_FK	cod_sup_FK

ER / RELACIONAL

Passo 5

Para cada **Relacionamento R de N:N** no esquema ER,

- criar uma nova relação para representar R,
- incluir como chave estrangeira as chaves primárias das relações que participam em R, estas chaves combinadas formarão a chave primária da relação,
- incluir também eventuais atributos de R.

TRAB_EM	cod_fun_FK	cod_proj_FK	horas

Passo 6

Para cada **Atributo Multivalorado A**, criar uma nova relação R, incluindo um atributo correspondendo a A mais a chave primária K da relação que tem A como atributo.

A chave primária de R é a combinação de A e K.

LOCAIS	cod_dep_FK	local

ESQUEMA RELACIONAL

FUNCIONÁRIO	cod	nome	sexo	salario		
DEPENDENTE	cod_FK	nome	dt_nasc	grau_p		
DEPARTAMENTO	cod	nome	cod_gerente_FK	dt_ini		
FUNCIONÁRIO	cod	nome	sexo	salario	cod_dep_FK	cod_sup_FK
TRAB_EM	cod_fun_FK	cod_proj_FK	horas			
LOCAIS	cod_dep_FK	local				

Um Esquema Relacional pode ser facilmente derivado de um Esquema Conceitual desenvolvido usando o Modelo E-R

ER / RELACIONAL

Para efetuar o mapeamento, devemos seguir um conjunto de regras levando em conta características específicas como força de uma Entidade, cardinalidade de um Relacionamento e tipo do Atributo.

- Entidade Regular
- Entidade Fraca
- Relacionamento 1:1
- Relacionamento 1:N
- Relacionamento N:M
- Atributo Multivalorado

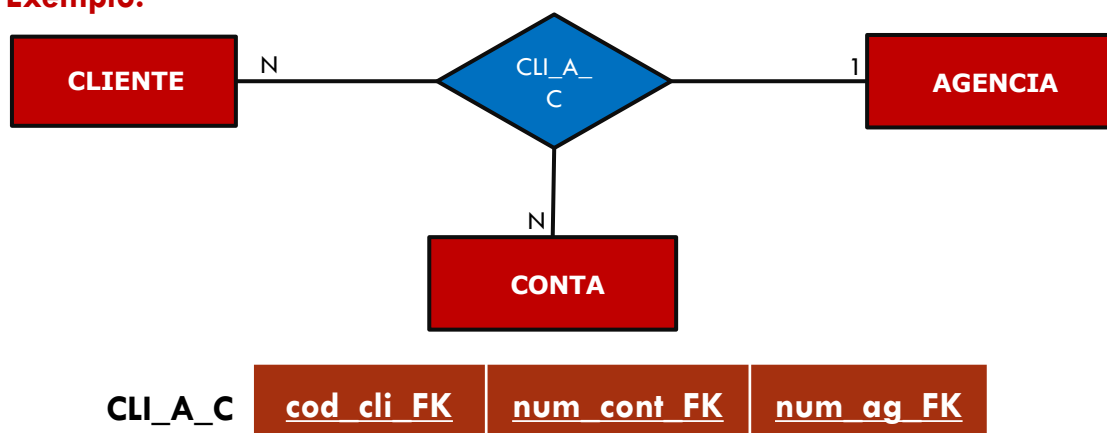
ER / RELACIONAL

Passo 7

Para cada **Relacionamento n-ário R (n > 2),**

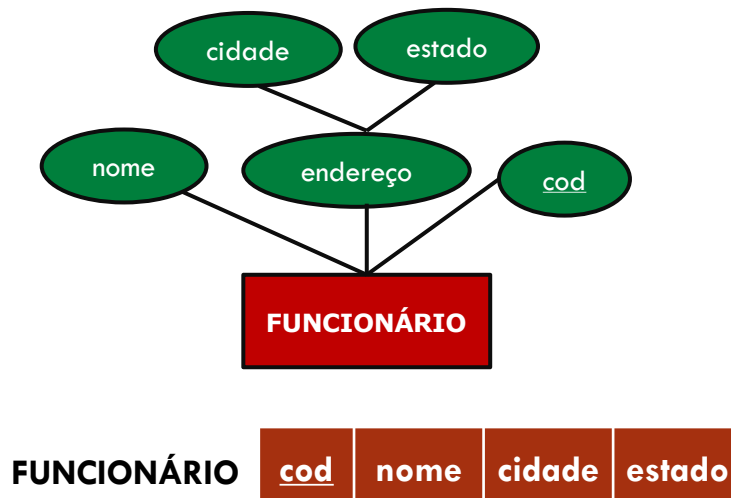
- criar uma nova relação S para representar R,
- incluir como chaves estrangeiras as chaves primárias das relações que representam as entidades participantes,
- incluir os eventuais atributos de R

A chave primária de S é normalmente a combinação das chaves estrangeiras.

ER / RELACIONAL

Passo 8

Para cada **Atributo Composto C** no esquema E-R que tenha como entidade proprietária E,

- Incluir todos atributos pertencentes a C como atributos da relação R, criada para representar E

Exemplo:

ER / RELACIONAL

Passo 9

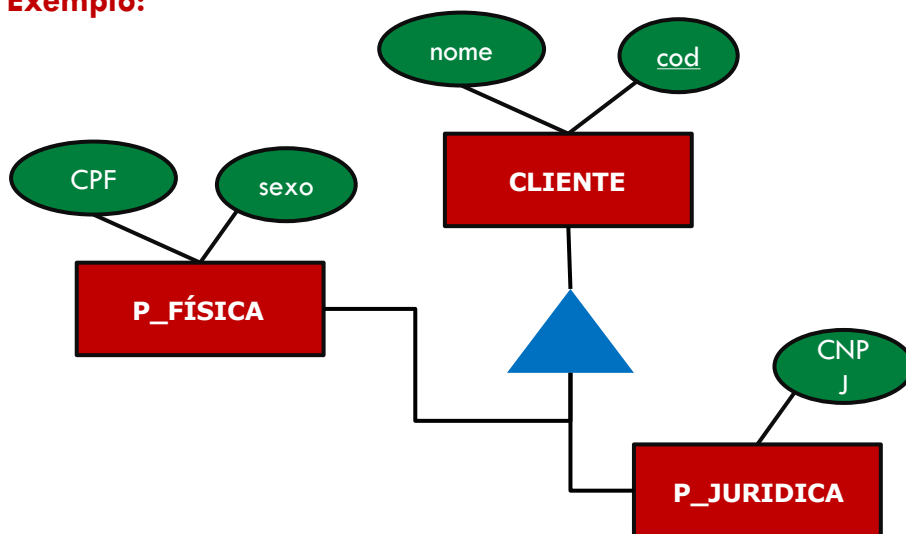
a) Para cada Entidade Regular G de **nível superior** que inclui **Generalização** no esquema ER

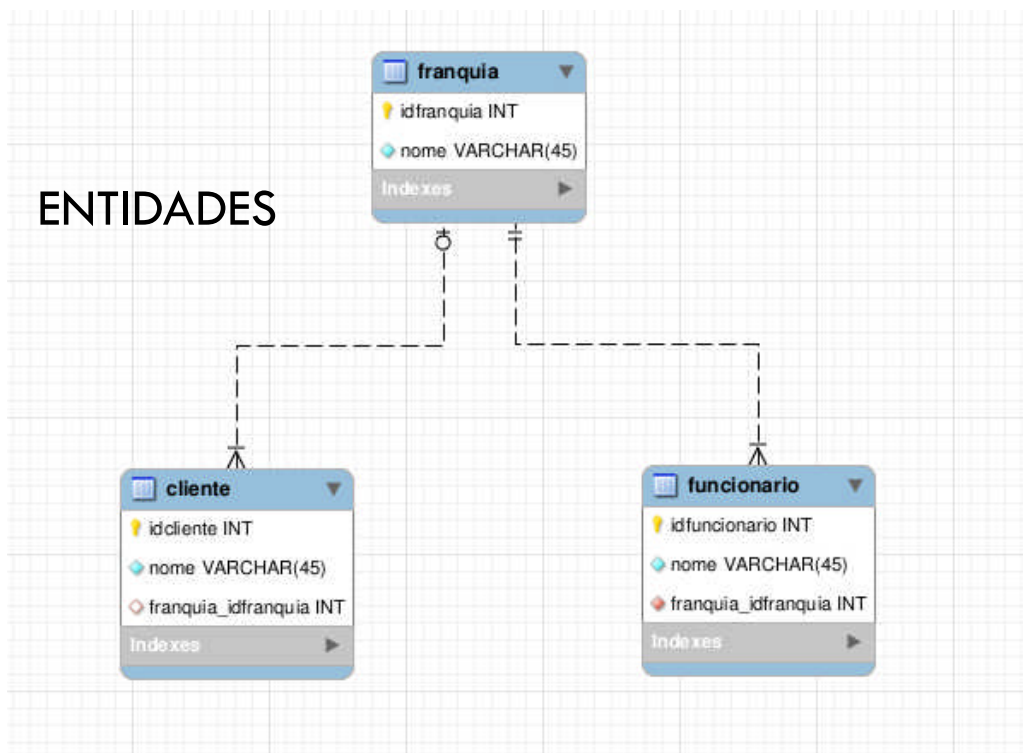
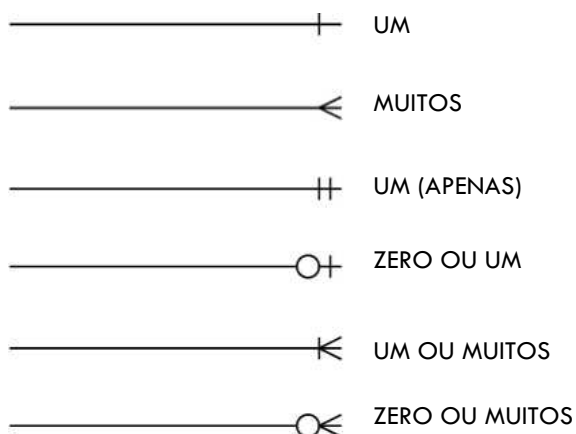
- criar uma relação R que inclui todos os atributos de R.

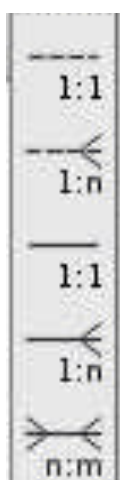
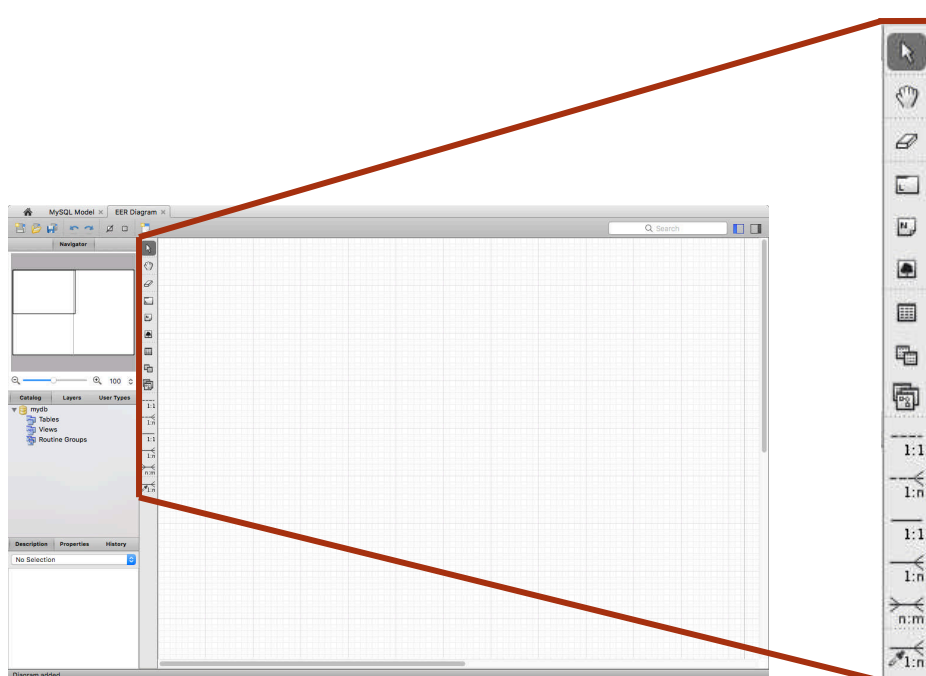
b) Para cada **Entidade Regular E** de **nível inferior** que inclui **Especialização** no esquema ER

- criar uma nova relação S para representar E,
- incluir os atributos de S

A **chave primária** da entidade de nível superior se torna atributo de **chave primária** de cada entidade de nível inferior.
Além disso, a **chave primária** da entidade de nível superior se torna atributo de ***chave estrangeira*** de cada entidade de nível inferior.

HERANÇA

CLIENTE	cod	nome	
P_FÍSICA	cod_cli_FK	sexo	CPF
P_JURIDICA	cod_cli_FK	CNPJ	

DER NOTAÇÃO DE JAMES MARTIN

CARDINALIDADES

MYSQL WORKBENCH

- Um relacionamento **com identificação** é aquele que é representado por uma chave estrangeira que é parte da composição da chave primária da tabela referenciada.
- Um relacionamento **sem identificação** é um relacionamento mais comum.

PARTE IV

NORMALIZAÇÃO DE BANCO DE DADOS

ROTEIRO

- Introdução e Problema de Tabelas Mal formadas
- Dependência Funcional
- Normalização
- As três primeiras Formas Normais (modelo relacional)
- Exemplos

INTRODUÇÃO

- Um dos objetivos principais do gerenciamento de Bancos de Dados é manter a **integridade** dos dados nele armazenados, e para esse fim, algumas **regras precisam ser consideradas**.
- Algumas dessas regras são garantidas pelo próprio gerenciador
  - tais como unicidade da chave, ligação entre relações através da chave estrangeira, etc.
- Outras regras são definidas nos programas de aplicação que ficam responsáveis por mantê-las.
- É necessário que as relações sejam bem fundamentadas, no sentido de **evitar redundâncias** que possam gerar, entre outros problemas, **inconsistência** de dados.
- Para procurar garantir esse aspecto, foi desenvolvida uma técnica chamada **Normalização**.
- O conceito de normalização foi introduzido por E. F. Codd em 1972.

INTRODUÇÃO

Inicialmente Codd criou as três primeiras formas de normalização chamando-as de: primeira forma normal (1NF), segunda forma normal (2NF) e terceira forma normal (3NF). Uma definição mais forte da 3NF foi proposta depois por Boyce-Codd, e é conhecida como forma normal de Boyce-Codd (FNBC).

Através do processo de normalização pode-se, gradativamente, substituir um conjunto de entidades e relacionamentos por um outro, o qual se apresenta "purificado" em relação às anomalias de atualização (inclusão, alteração e exclusão).

Normalização de relações é portanto uma técnica que permite depurar um projeto de banco de dados, através da identificação de inconsistências (informações em duplicidade, dependências funcionais mal resolvidas, etc).

À medida que um conjunto de relações passa para uma forma normal, vamos construindo um banco de dados mais confiável.

O objetivo da normalização não é eliminar todos as inconsistências, e sim controlá-las.

INTRODUÇÃO

- Redundância é a causa de vários problemas com esquemas relacionais:
  - armazenamento redundante, anomalias de inserção, de exclusão e de atualização.
- Restrições de integridade podem ser usadas para identificar esquemas com esses problemas e para sugerir refinamentos.
- Principal técnica de refinamento: a **decomposição** de um esquema em sub esquemas.
- A decomposição deve ser usada cuidadosamente.
  - Há motivos para se decompor uma relação?
  - A decomposição pode causar problemas?

INTRODUÇÃO

- Considere o esquema:

Pacientes(Id, Nome, Endereço, Telefone, Sexo,
Data_nascimento, Sigla_convênio,
Nome_convênio, Endereço_convênio,
Telefone_convênio)

- Esse é um exemplo de um esquema mal projetado!
  - Por que?

Pacientes(Id, Nome, Endereço, Telefone, Sexo,
Data_nascimento, Sigla_convênio,
Nome_convênio, Endereço_convênio,
Telefone_convênio)

  - Por que os dados de um convênio (nome, endereço e telefone do convênio) são repetidos para cada paciente associado a esse convênio.
    - Por exemplo, os dados da UNIMED serão repetidos para cada um de seus associados.

INTRODUÇÃO

- **Anomalia de Inserção:**
  - Quando se inserir um paciente é preciso inserir também os dados do convênio, mesmo que já estejam cadastrados.
  - Não é possível inserir um convênio sem inserir também um paciente.
- **Anomalia de Exclusão:**
  - Ao se excluir um paciente, se este for o único associado de um convênio então os dados do convênio serão perdidos.
- **Anomalia de Modificação:**
  - Para se modificar os dados de um convênio, é preciso atualizar os mesmos dados em todas as tuplas de pacientes que estejam associados àquele convênio.

DEPENDÊNCIA FUNCIONAL

- Dependências funcionais (DFs) são restrições de integridade mais gerais que as restrições de chave.
- Exemplo de dependência funcional:

{Sigla_convênio} → {Nome_convênio, Endereço_convênio, Telefone_convênio}

- Leia-se: **Sigla_convênio** determina funcionalmente Nome_convênio, Endereço_convênio e Telefone_convênio.
- Significado: "Se duas linhas da tabela Pacientes tiverem o mesmo valor de **Sigla_convênio**, então elas tem de ter o mesmo valor de Nome_convênio, de Endereço_convênio e de Telefone_convênio".

DEPENDÊNCIA FUNCIONAL

- Tente identificar as dependências funcionais na tabela abaixo:

codigo	nome	cargo	salario
2	José	Engenheiro	2500.00
18	Juca	Programador	1300
15	Sócrates	Analista	2800
79	Platão	Professor	1200
46	Aristócrates	Programador	1300

{codigo} → {nome}
{cargo} → {salário}

DEPENDÊNCIA FUNCIONAL

- Identifique as dependências funcionais:

Num_NF	Cod_Produto	Descricao	Preco	Data	QTD
001	PC	Computador	2300.00	2019-02-12	2
001	IMP	Impressora	899.00	2019-02-12	1
002	PC	Computador	2300.00	2019-02-13	1
003	PC	Computador	2300.00	2019-02-14	3

{Num_NF} → {Data}
{Cod_Produto} → {Descricao, Preco}
{Num_NF, Cod_Produto} → {Quantidade}

DEPENDÊNCIA FUNCIONAL

- Uma restrição de chave é um caso especial de DF:
  - a chave determina funcionalmente todos os outros atributos da tabela.
- Como **Id** é chave da tabela Pacientes, temos que:

{Id} → {Nome, Endereco, Telefone, Sexo, Data_nascimento, Sigla_convenio, Nome_convenio, Endereço_convenio, Telefone_convenio}

- Certas DFs causam redundância!
  - Por exemplo → Para cada associado de um convênio, os dados do convênio são repetidos na tabela Pacientes.
- A causa desse problema é a DF.

{Sigla_convênio} → {Nome_convênio, Endereço_convênio, Telefone_convênio}

PROJETO DE BANCO DE DADOS

- O objetivo do projeto de um BD relacional
  - Gerar um conjunto de esquemas de relações que permitam armazenar informações sem redundância desnecessária
  - Recuperar informações facilmente

NORMALIZAÇÃO

- Esse processo é composto pelas chamadas formas normais.

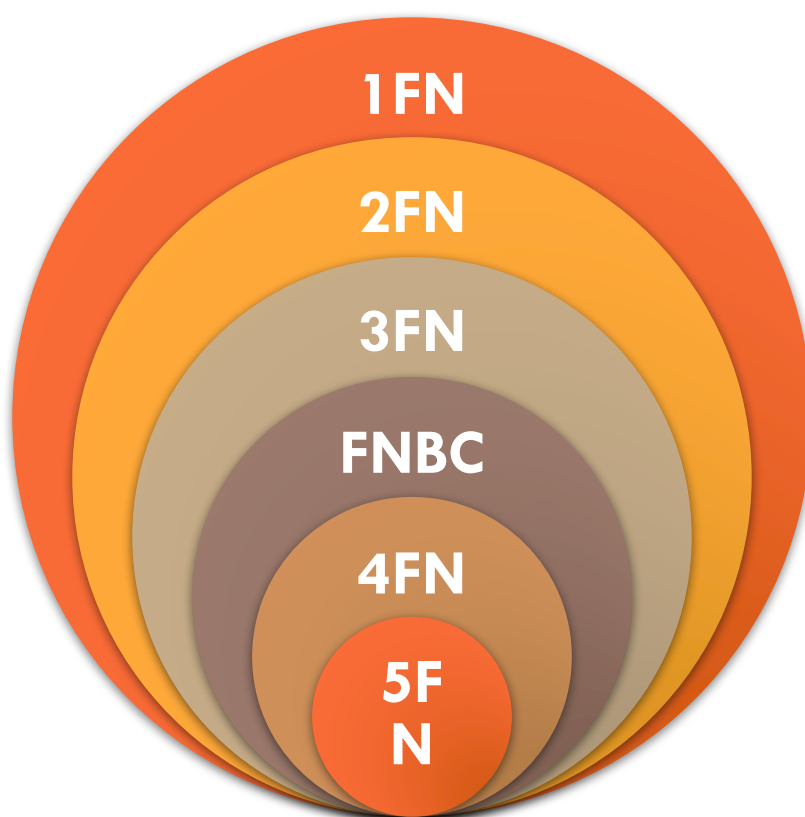
Um modelo de base de dados que respeite os princípios estipulados até à 3ª FN é considerado adequadamente elaborado para funcionar num SGBD relacional.

- 1ª Forma Normal (1ª FN)
- 2ª Forma Normal (2ª FN)
- 3ª Forma Normal (3ª FN)

- Forma normal de Boyce-Codd (FNBC)
- 4ª Forma Normal (4ª FN)
- 5ª Forma Normal (5ª FN)

NORMALIZAÇÃO

- Conjunto de regras que ajudam na definição de bancos de dados que não contenham redundância desnecessária e que permitam o fácil acesso às informações.
- Método permitindo identificar a existência de problemas (anomalias) **n** projeto de um BD relacional.
- Converte progressivamente uma tabela em tabelas de grau e cardinalidade menores até que pouca ou nenhuma redundância de dados exista
  - Há diferentes níveis de normalização, de acordo com as condições atendidas
  - A hierarquia entre as formas normais indica que uma tabela só pode estar numa forma mais avançada se, além de atender as condições necessárias, já estiver na forma normal imediatamente anterior

NÍVEIS NORMALIZAÇÃO

NORMALIZAÇÃO

- Se a normalização é bem sucedida:
  - O espaço de armazenamento dos dados diminui
  - A tabela pode ser atualizada com maior eficiência
  - A descrição do BD será imediata

- A finalidade das regras de normalização é evitar anomalias de atualização no banco de dados
- **Anomalias de inserção**
  - Evitar a repetição desnecessária de dados (redundância)
- **Anomalias de alteração**
  - Evitar inconsistências e reduzir o esforço para a atualização dos dados
- **Anomalias de exclusão**
  - Evitar a perda de informações associadas a um dado registro

NORMALIZAÇÃO

- Considere uma única tabela Vendas para representar as informações sobre os negócios de uma Livraria:

nome_cliente	cod_livro	titulo	autor	preco	data_compra
Platão	353890	O poder da ação	Paulo Vieira	29.90	2018-11-23
...	...	...	...	...	...
Aristóteles	538309	Geração de Valor	Flávio Augusto	24.90	2019-01-16

- Caso fosse preciso registrar a compra de 5 Livros iguais para um mesmo cliente, as seguintes anomalias seriam observadas:
  - **Anomalia de inserção:** Redundância em todas as colunas (5 linhas iguais na tabela).
  - **Anomalia de alteração:** A mudança no preço do Livro deveria ser feita em todas as linhas correspondentes da tabela.
  - **Anomalia de exclusão:** Só haveria registro dos Livros que fossem comprados; se a única venda de um Livro fosse apagada, não haveria mais informações sobre aquele Livro.

PRIMEIRA FORMA NORMAL (1FN)

- **Conceito:** Uma variável de relação (tabela) está em 1FN se, e somente se, em todo valor válido dessa variável de relação, cada tupla contém exatamente um valor para cada atributo
- Os atributos devem ser atômicos (indivisíveis)
- Atributos compostos ou multivalorados devem ser representados por novas linhas ou novas tabelas

- Exemplo: Tabela Controle de Faltas numa Escola
- A tabela abaixo não está na 1FN

cod_turma	aluno	professor	sala	capacidade	qtd_faltas
BD2019	João Pedro Maria Clara José Costa	Peter Carlos	202	50	2 0 5
...	...	...	...	...	...

- Os atributos **aluno** e **qtd_faltas** não são atômicos (há mais de um valor para cada registro)

PRIMEIRA FORMA NORMAL (1FN)

- Passos para obtenção da 1FN em uma tabela:
  1. Identificar a chave primária da tabela.
  2. Identificar os atributos compostos ou multivalorados.
  3. Incluir uma coluna/linha para cada atributo composto/multivalorado.

- A tabela abaixo está na 1FN (atributos atômicos)

cod_turma	aluno	professor	sala	capacidade	qtd_faltas
BD2019	João Pedro	Peter Carlos	202	50	2
BD2019	Maria Clara	Peter Carlos	202	50	0
BD2019	José Costa	Peter Carlos	202	50	5
...	...	...	...	...	...

- O próximo passo é observar se ela está também na 2FN

SEGUNDA FORMA NORMAL (2FN)

Conceito 1: uma variável de relação está em 2FN se, e somente se, ela está em 1FN e todo atributo não-chave é irredutivelmente dependente da chave primária.

Conceito 2: uma variável de relação está em 2FN se, e somente se, ela está em 1FN e, para tabelas com chave primária composta, cada coluna não-chave depende de toda a chave, e não de apenas uma parte dela.

Dica: tabelas em 1FN e com Chave Primária simples estão automaticamente em 2FN.

- A tabela abaixo está na 1FN mas não está na 2FN

cod_turma	aluno	professor	sala	capacidade	qtd_faltas
BD2019	João Pedro	Peter Carlos	202	50	2
BD2019	Maria Clara	Peter Carlos	202	50	0
BD2019	José Costa	Peter Carlos	202	50	5
...	...	...	...	...	...

- Os atributos **professor**, **sala** e **capacidade** dependem apenas de **cod_turma** (repetição para todos os alunos da turma)

SEGUNDAFORMA NORMAL (2FN)

Passos para obtenção da 2FN em uma tabela

1. Deixá-la em 1FN
2. Identificar os atributos que não fazem parte da chave primária da tabela
3. Para cada um desses atributos, analisar se seu valor é determinado por parte ou pela totalidade da chave
4. Criar novas tabelas para os atributos parcialmente dependentes, incluindo a parte da chave correspondente, e retirá-los da tabela original

- **As seguintes tabelas estão na 2FN:**

cod_turma	aluno	qtd_faltas
BD2019	João Pedro	2
BD2019	Maria Clara	0
BD2019	José Costa	5
...	...	...

cod_turma	professor	sala	capacidade
BD2019	Peter	202	50
WEB2019	Mauricio	202	50
LTP2019	Bil	202	50
...	...	...	...

TERCEIRA FORMA NORMAL (3FN)

Conceito 1: uma variável de relação está em 3FN se, e somente se, ela está em 2FN e todo atributo não-chave é dependente de forma não transitiva da chave primária.

Conceito 2: uma variável de relação está em 3FN se, e somente se, ela está em 2FN e todo atributo não-chave depende apenas da chave, e não de outros atributos não-chave.

Dica: tabelas em 2FN e com nenhum ou um atributo além da chave estão automaticamente em 3FN.

- A seguinte tabelas estão na 2FN, mas não está em 3FN

cod_turma	professor	sala	capacidade
BD2019	Peter	202	50
WEB2019	Mauricio	202	50
LTP2019	Bil	202	50
...	...	...	...

- O atributo **capacidade** depende do atributo **sala**, e não da chave **cod_turma**

TERCEIRA FORMA NORMAL (3FN)

- Passos para obtenção da 3FN em uma tabela
  1. Deixá-la em 2FN
  2. Identificar os atributos que não participam da chave primária da tabela
  3. Para cada um desses atributos, analisar se seu valor é determinado por algum outro atributo não pertencente à chave primária
  4. Criar novas tabelas para os atributos que não dependem exclusivamente da chave, incluindo o atributo determinante correspondente, e retirá-los da tabela original

- As seguintes tabelas estão na 3FN

cod_turma	professor	sala_FK
BD2019	Peter	202
WEB2019	Mauricio	202
LTP2019	Bil	202
...	...	...

cod_turma	aluno	qtd_faltas
BD2019	João Pedro	2
BD2019	Maria Clara	0
BD2019	José Costa	5
...	...	...

sala	capacidade
202	50
102	40
302	50
...	...

REGRAS GERAIS – NORMALIZAÇÃO

- **1FN:** Eliminar atributos multivalorados ou compostos.
- **2FN:** Eliminar atributos que dependem apenas de parte da chave primária composta.
- **3FN:** Eliminar atributos que dependem de atributos não-chave.

NORMALIZAÇÃO

- A normalização serve para medir o quanto uma tabela é melhor do que outra, baseada em alguma teoria. Podemos também pegar um banco de dados já modelado e desejar que este seja melhor refinado.
- Aumentar o nível de normalização contribui para melhorar a qualidade do projeto do banco de dados.
- Há ainda outras três formas normais (FNBC, 4FN e 5FN), cada uma com suas restrições.
- Essas outras formas normais não serão vistas nessa nossa jornada.

EXERCÍCIO #01

- A tabela a seguir representa as vendas numa loja de CDs.
- Considerando as formas normais vista (1FN, 2FN e 3FN), indicar quais são atendidas pelo projeto.
- Caso alguma delas não seja atendida, identifique o problema e proponha as mudanças necessárias.

TABELA DE VENDAS

Chave composta

cliente	cod_cd	cantor	musica	duracao	preco	data_compra
Atena	238998 898274	Ferrugem Melim	Pirata e Tesouro Eu juro Meu Abrigo Ouvi Dizer	00:03:17 00:03:27 00:03:53 00:03:47	19.90 23.90	2019-02-12
...	...	...	...	...	...	...

Dica: A solução tem 4 tabelas

EXERCÍCIO #02

- A tabela abaixo representa os pedidos de produtos de software para uma loja e não obedece nenhuma das formas normais vistas (1FN, 2FN e 3FN).
- Indique os passos para deixá-la em cada uma dessas formas normais

TABELA DE PEDIDOS

num_pedido	data	fornecedor	cnpj	endereco	cod_produto	nome	qtd	preco
003	2019-01-03	HouseSoft	010299	R. 13 de Mario, 27	033A 002M 145J	Office 2016 Corel Photoshop	3 4 2	140.00 699.00 897.00
004	2019-02-08	BrazilSoft	020300	Av. Catu, 189	002M 146J 034A	Corel Illustrator MS Project	3 1 10	699.00 897.00 386.00
...	...	...	...	...	...	...		

Dica: A solução tem 4 tabelas

SOLUÇÃO: EXERCÍCIO #01 (TABELA DE VENDAS)

- A tabela Vendas não está na **1FN**, pois há vários atributos não atômicos
- Para deixá-la em **1FN**, é preciso dividir esse atributos em linhas

cliente	cod_cd	cantor	musica	duracao	preco	data_compra
Atena	238998	Ferrugem	Pirata e Tesouro	00:03:17	19.90	2019-02-12
Atena	238998	Ferrugem	Eu juro	00:03:27	19.90	2019-02-12
Atena	898274	Melim	Meu Abrigo	00:03:53	23.90	2019-02-12
Atena	898274	Melim	Ouvi Dizer	00:03:47	23.90	2019-02-12
...	...	...	...	...	...	...

SOLUÇÃO: EXERCÍCIO #01 (TABELA DE VENDAS)

- A tabela Vendas não está na **2FN**, pois há atributos que dependem apenas de parte da chave primária composta.
- Para deixá-la em **2FN**, é preciso cria uma nova tabela.

TABELA VENDAS

cliente	cod_cd_FK	data_compra
Atena	238998	2019-02-12
Atena	898274	2019-02-12

TABELA CDS

cod_cd	cantor	preco
238998	Ferrugem	19.90
898274	Melim	23.90

TABELA MUSICAS

musica	duracao
Pirata e Tesouro	00:03:17
Eu juro	00:03:27
Meu Abrigo	00:03:53
Ouvi Dizer	00:03:47

TABELA MÚSICAS_CD

cod_cd_FK	musica_FK
238998	Pirata e Tesouro
238998	Eu juro
898274	Meu Abrigo
898274	Ouvi Dizer

SOLUÇÃO: EXERCÍCIO #01 (TABELA DE VENDAS)

- As tabelas já estão na 3FN, pois não há atributos que dependem de atributo não-chave.
- Caso não estivessem na 3FN, para deixá-las em 3FN, seria preciso criar uma nova tabela.

AGRADECIMENTOS

Obrigado por adquirir este livro, espero que tenha contribuído para o seu crescimento profissional!

Não esqueça de me acompanhar nas redes sociais, saiba mais em

Não esqueçam de me seguir nas redes sociais e de se inscrever no canal.

Facebook: http://facebook.com.br/tassiogoncalvesg

Instagram: http://instagram.com/tassiogoncalvesg

LinkedIn: https://www.linkedin.com/in/tassiogoncalvesg/

YouTube: https://youtube.com/c/TássioGonçalves/

Acesse material de cursos e artigos gratuitos no meu site: https://tassiogoncalves.com.br